

REMARKS

This amendment is submitted in response to the Final Office Action dated November 15, 2006. Reconsideration and allowance of the claims are requested.

In the Office Action, claims 1-10 and 17-27 were examined. Claims 1-3, 10 and 17 were rejected under 35 U.S.C. 102(b) as anticipated by Boucher (U.S. 6,334,153). Claims 4-9, and 18-27 were rejected under 35 U.S.C. 103 as unpatentable over Boucher and further in view of Adams (U.S. 6,775,693). These rejections are respectfully traversed.

To limit the issues after final rejection, claims 17-20 have been cancelled; applicant retains the right to present these claims in a continuation application. Claims 10 and 23 are combined as claim 10 to emphasize the distinctions spelled out below. Claim 24 has been amended to change its dependency to claim 10, and claim 27 has been amended to correct an informality.

As explained at length in the pending application, by carefully dividing TCP/IP processing task between hardware and software and specifically delegating key repetitive portions of these tasks to the hardware, and by optimizing the software to take full advantage of these capabilities, the system cost is kept low while achieving high throughput and low CPU utilization. Although the application as presented describes a number of features which are used to implement this approach, the present claims in this application emphasize the dual track nature of the approach to uploading frame data that is transmitted for TCP processing. As specifically recited in the independent claims 1 and 10, a hardware subsystem is configured to process frames of certain connections delegated by a TCP stack to produce frame data and be stored at the direction of the hardware in system memory. Under certain circumstances, the hardware subsystem will store the frame data in a legacy buffer for later processing in a central processing unit.

The system memory comprises a connection table for storing data and tracking the active connections within the system including the delegated connections. As disclosed and claimed, the hardware creates the necessary connection table (CT) entry,

and the hardware maintains the CT entry point for each received packet. When the hardware needs to access a CT entry for a connection, it simply uses the contents of the packet TCP destination port field as a direct index into the CT. No hash functions or table management functions which would incur software utilization and time are needed. To avoid synchronization issues between hardware and software, the reads and writes to CT entries and system memory are made through the hardware.

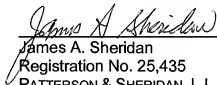
The features described above are not taught by the Boucher reference. The Examiner cites column 45, lines 16-28 of Boucher as teaching a system memory including a connection table (CT) for storing data for active connection. However, claim 1 and others further require that the reads and writes to CT entries are made through the hardware. In contrast, in Boucher, as spelled out beginning at column 44, a routing table is maintained using an algorithm known as PATRICIA which, the inventor points out, is a complicated algorithm that is costly to set up. Therefore, rather than use the hardware implemented approach of the present invention, as set forth in claim 1, and further emphasized in claims 3, 4, 6, 8 and 9, the Boucher reference utilizes a software driven and controlled system for establishing and maintaining the CT entries. Therefore, all of these claims must be allowed over the references of record.

As to the combination of Boucher and Adams which is used to reject most claims of the present application, the Examiner at paragraph 10 of the Office Action concedes that Boucher does not teach a system wherein the hardware is configured to process frames to produce partially processed frame data, and further to upload the partially processed framed data to a portion of system memory. To overcome this deficiency, the Examiner relies on Adams, especially at column 8, lines 38-45 and column 4, lines 5-10. A close reading of Adams, both the cited portions and the remainder of his disclosure, however, fails to reveal any such reliance on hardware. In fact, at column 2, lines 34-41, Adams teaches that the basics of his system include a transport process that create a "virtual circuit" and defines a transport layer that implements a flow control mechanism and an error control mechanism. Clearly a person of skill in the art, following Adams, would implement the flow control and error checking in software, as taught, rather than in hardware. In his detailed description of his invention, Adams does not teach a modification to this transport layer approach (see also column 3, lines 13-24).

Rather, he builds upon the software approach by defining a maximum transfer unit (MTU) transferred through a network including a fiber channel network. At the cited portion of column 4, lines 5-10, Adams only teaches that, once data has been segmented into MTU's, they are to be transferred over a specific type of fiber channel network. The Examiner further relies on column 8, lines 38-45. The features discussed in this section, however, while vague and difficult to relate to the present invention, clearly appear to be driven by drivers 125/135 which are implemented in software (see column 5, lines 11-30, which describes defining the scope of the functions being carried out during boot-up, which of course occurs under software control).

In view of these clear distinctions, and the combination of the claims 10 and 23 to emphasize the hardware driven nature of the present invention, reconsideration and allowance of the claims are respectfully requested.

Respectfully submitted,


James A. Sheridan
Registration No. 25,435
PATTERSON & SHERIDAN, L.L.P.
3040 Post Oak Blvd. Suite 1500
Houston, TX 77056
Telephone: (713) 623-4844
Facsimile: (713) 623-4846
Attorney for Applicants